Database Systems

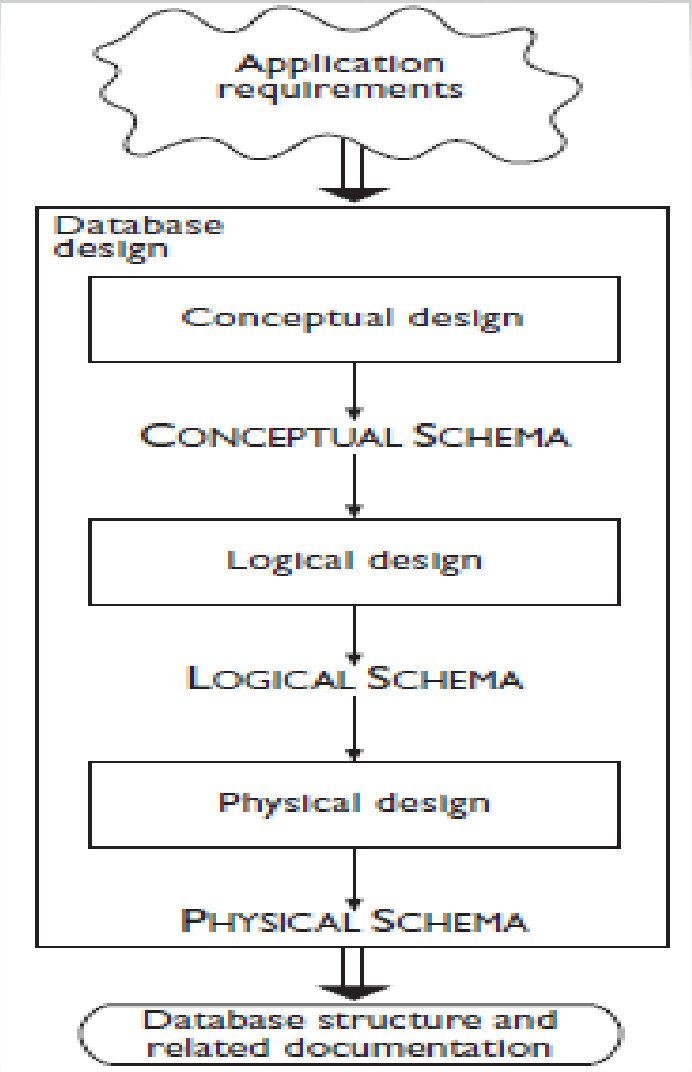
Lecture # 9

Dr Sherin ElGokhy

A Database design methodology

- Within the field of databases, a design methodology has been consolidated over the years.
- It is based on a simple but highly efficient engineering principle: separate the decisions relating to ‘what’ to represent in the database, from those relating to ‘how’ to do it.
- This methodology is divided into three phases to be followed consecutively.

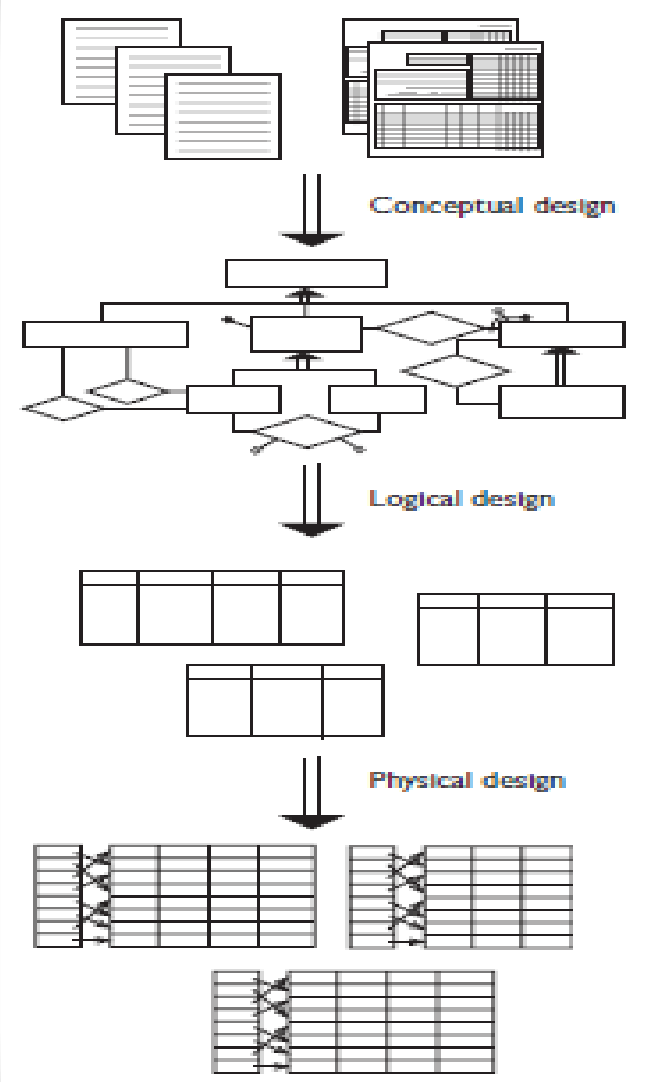
The phases of database design



Phases of database design

- **Conceptual design.** The purpose of this is ***to represent the informal requirements of an application*** in terms of a conceptual schema that ***refers to a conceptual data model***.
- **Logical design.** This consists of the translation of the conceptual schema defined in the preceding phase, into the logical schema of the database that ***refers to a logical data model.***
- **Physical design.** In this phase, the logical schema is completed with the details of ***the physical implementation*** (file organization and indexes) on a given DBMS. The product is called the physical schema and ***refers to a physical data model.***

The products of the various phases of a relational database with the E-R model



Database Systems (P. Atzeni, S. Ceri, S. Paraboschi and R. Torlone)

Two main types of models

Models

Logical models

- used in DBMSs for the organization of data at a level that although abstract from physical structures, it reflects a particular organization
- examples: **relational, network, hierarchical, object**

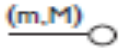
Conceptual models

- used to describe data in a way that is completely independent of any system, with the goal of representing the concepts of the real world rather than the data needed for their representation; they are used in the early stages of database design
- the most popular is the **Entity-Relationship(E-R)** model

The Entity Relationship model

- **The Entity-Relationship (E-R)** model is a conceptual data model, that provides a series of constructs capable of describing the data requirements of an application in a way that is easy to understand and is independent of the criteria for the management and organization of data on a database system.
- For every construct, there is a corresponding graphical representation. This representation allows us to define an E-R schema diagrammatically.

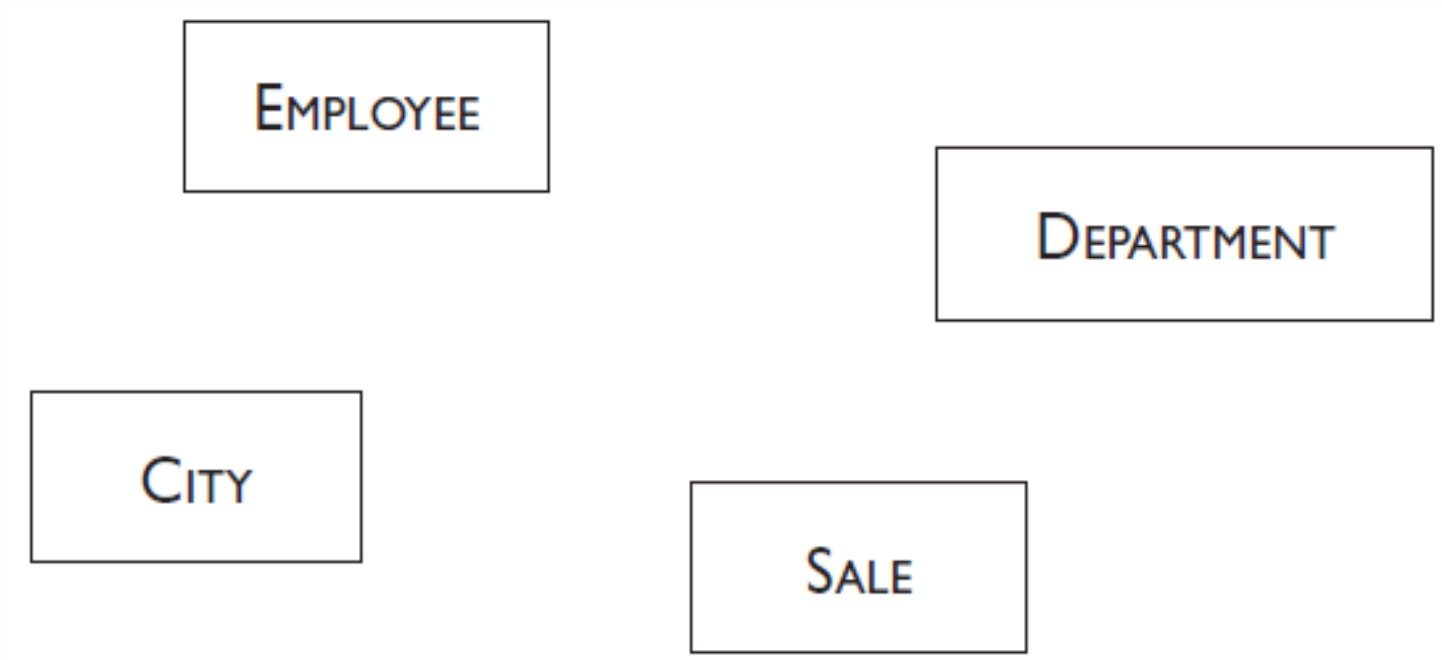
The constructs of the E-R model and their graphical representation

Construct	Graphical representation
Entity	
Relationship	
Simple attribute	
Composite attribute	
Cardinality of a	
Cardinality of an attribute	
Internal identifier	
External identifier	
Generalization	
Subset	

Entities

- **These represent classes of objects (facts, things, people, for example) that have properties in common and an autonomous existence.**
- **CITY, DEPARTMENT, EMPLOYEE, PURCHASE and SALE are examples of entities in an application for a commercial organization.**
- **An occurrence of an entity is an object of the class that the entity represents.**

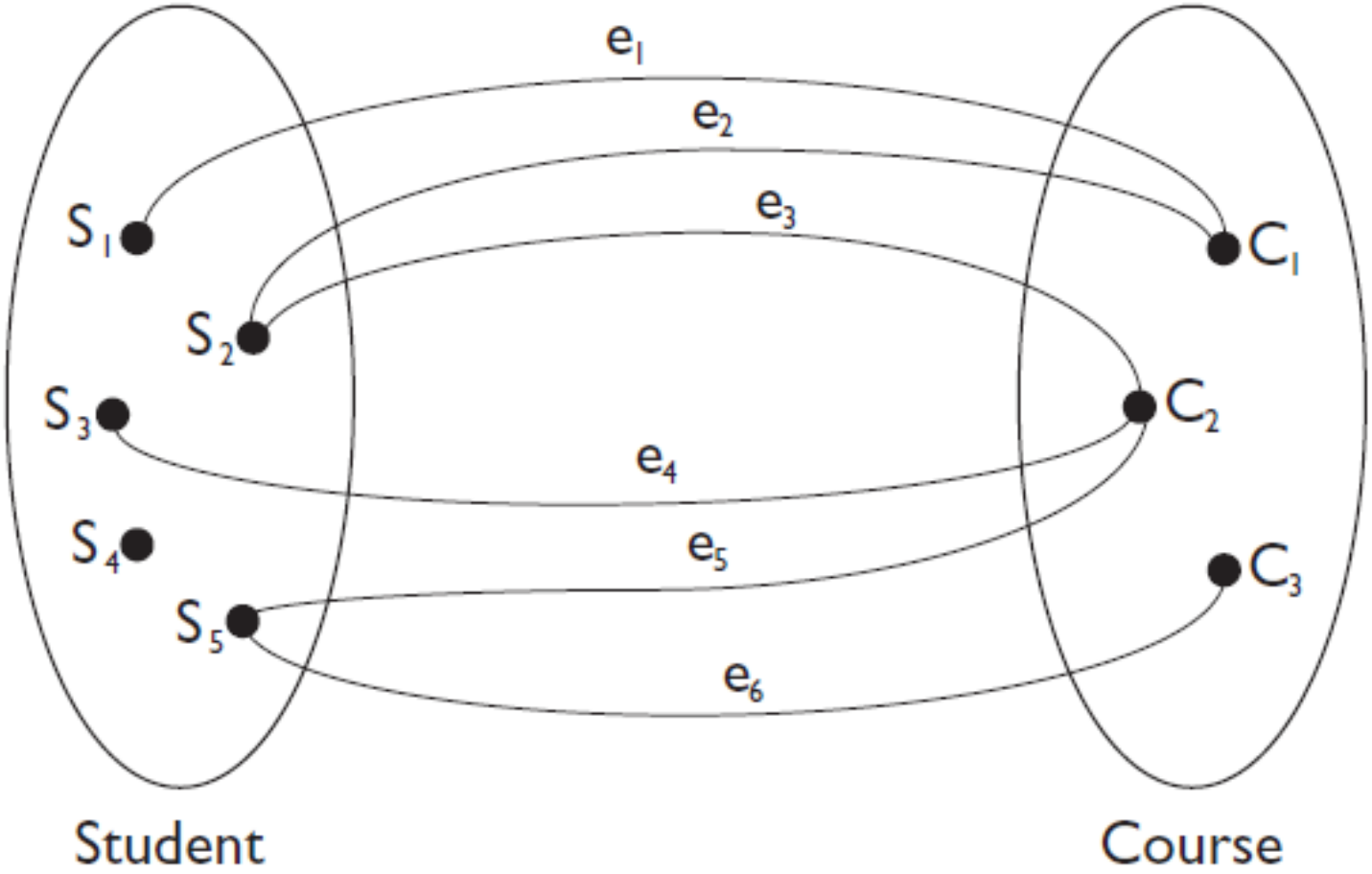
Examples of entity of the E-R model



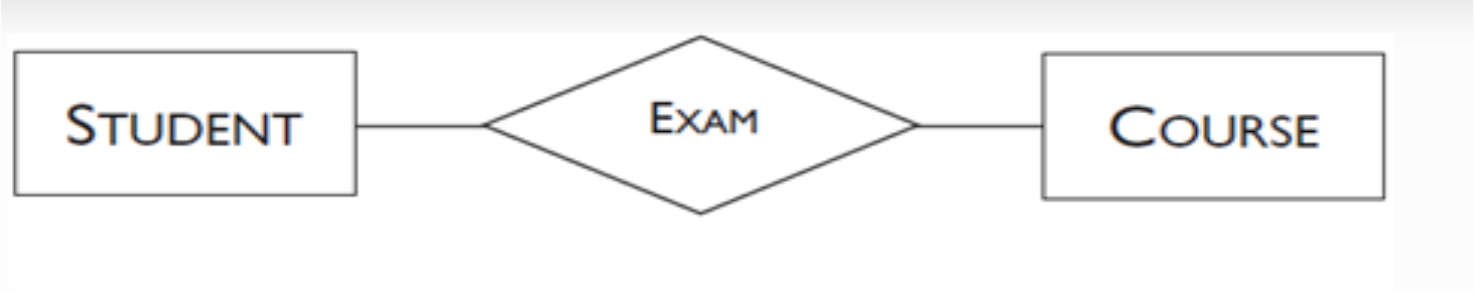
Relationships

- They represent logical links between two or more entities.
- **RESIDENCE** is an example of a relationship that can exist between the entities **CITY** and **EMPLOYEE**; **EXAM** is an example of a relationship that can exist between the entities **STUDENT** and **COURSE**.
- An occurrence of a relationship is an n-tuple made up of occurrences of entities, one for each of the entities involved.

Example of occurrences of the EXAM relationship

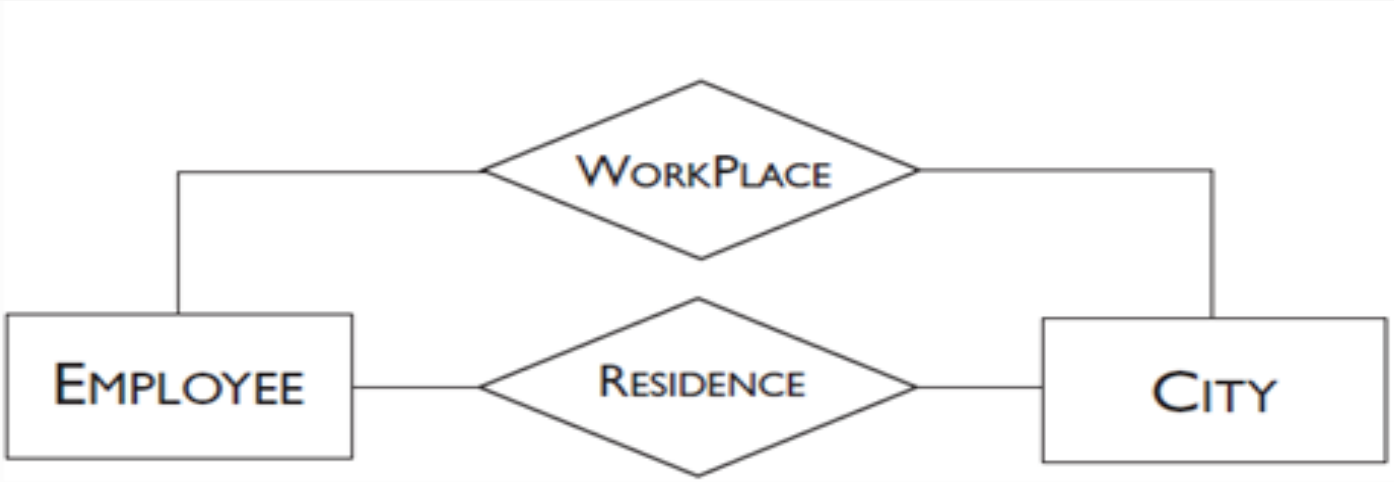


Example of relationships in the E-R model



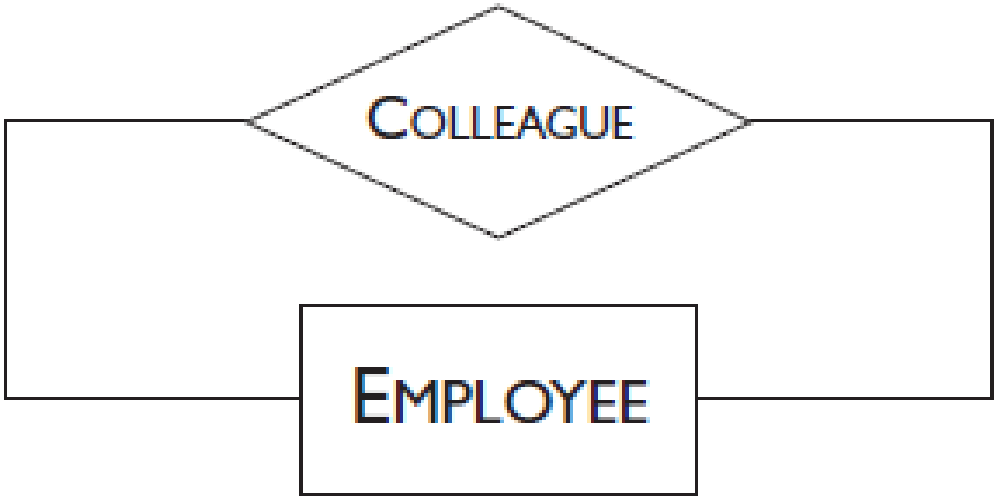
A relationship EXAM defined on the two entities STUDENT and COURSE

Example of relationships in the E-R model



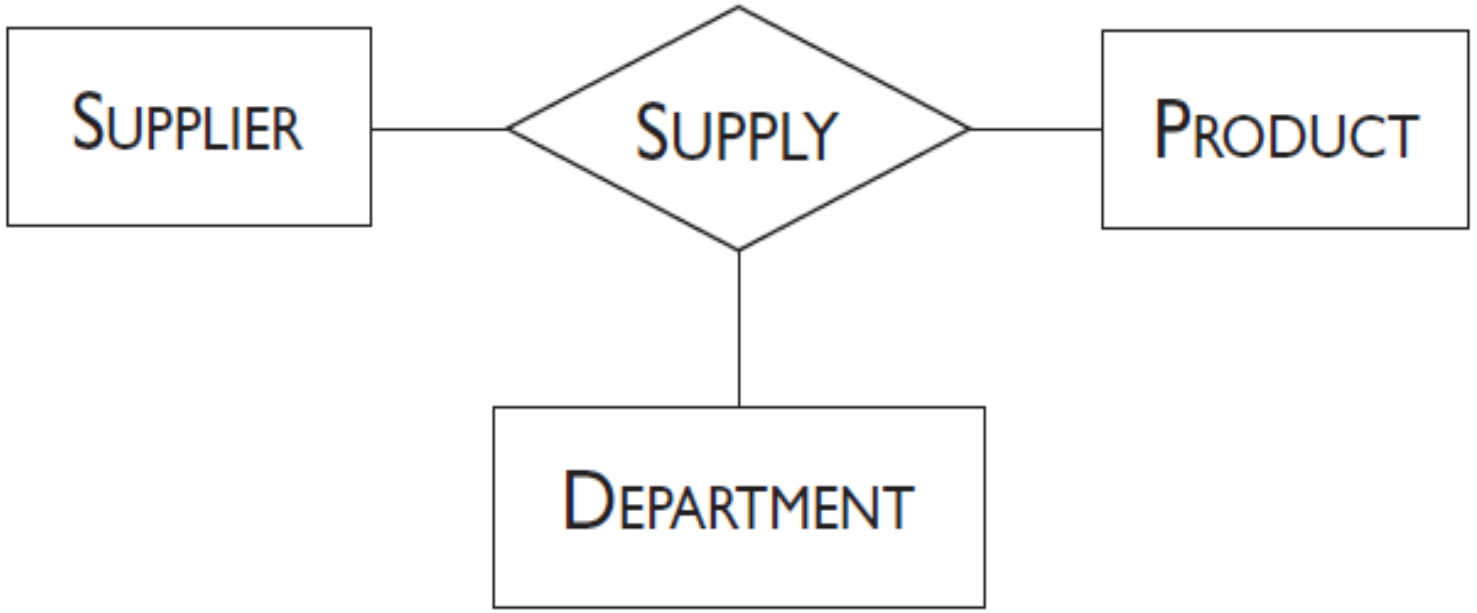
Two different relationships defined on the same pair of entities

Examples of recursive relationships in the E-R model



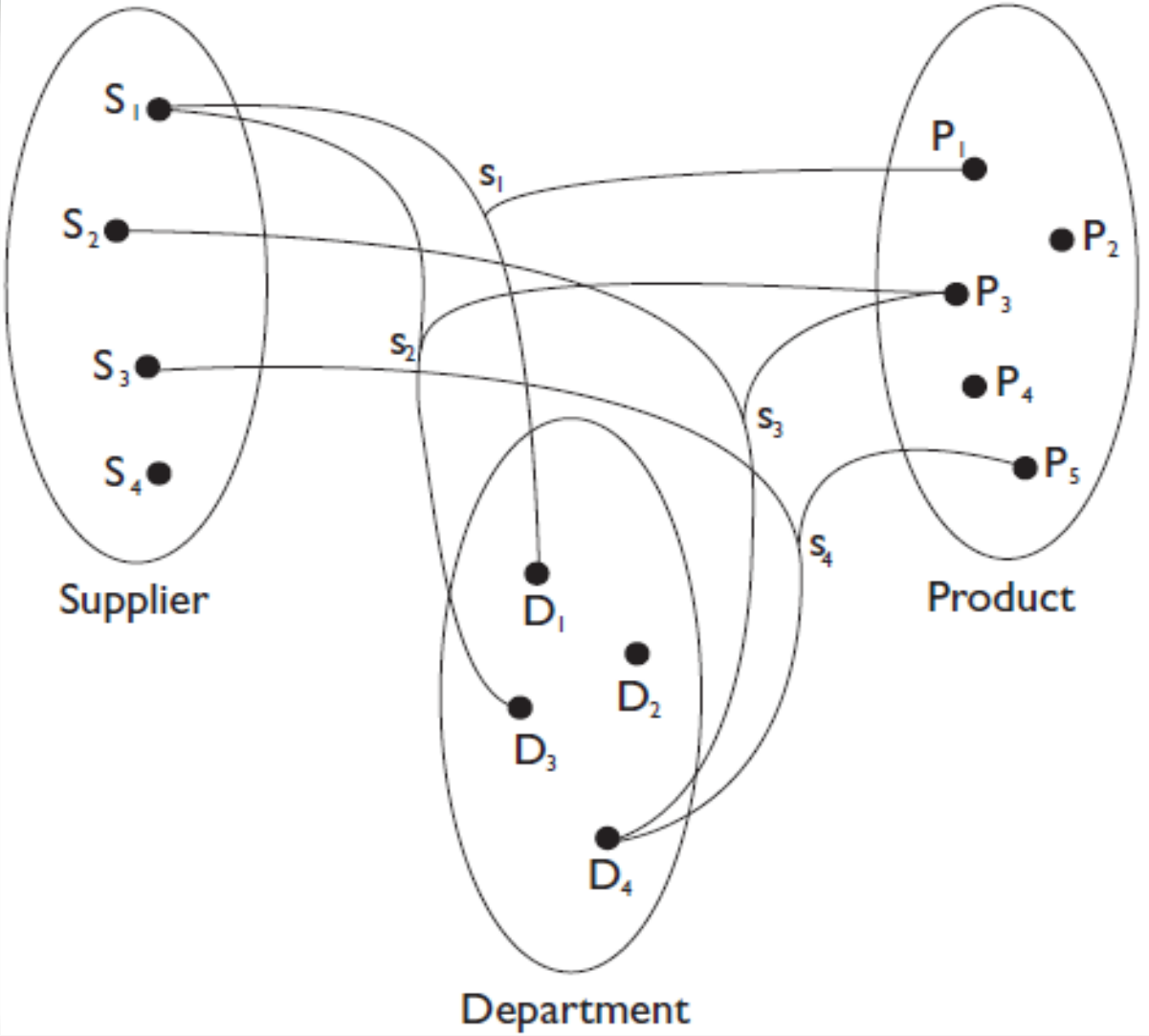
A relationship between an entity and itself

Example of a ternary relationship in the E-R model



Relationship defined on three entities

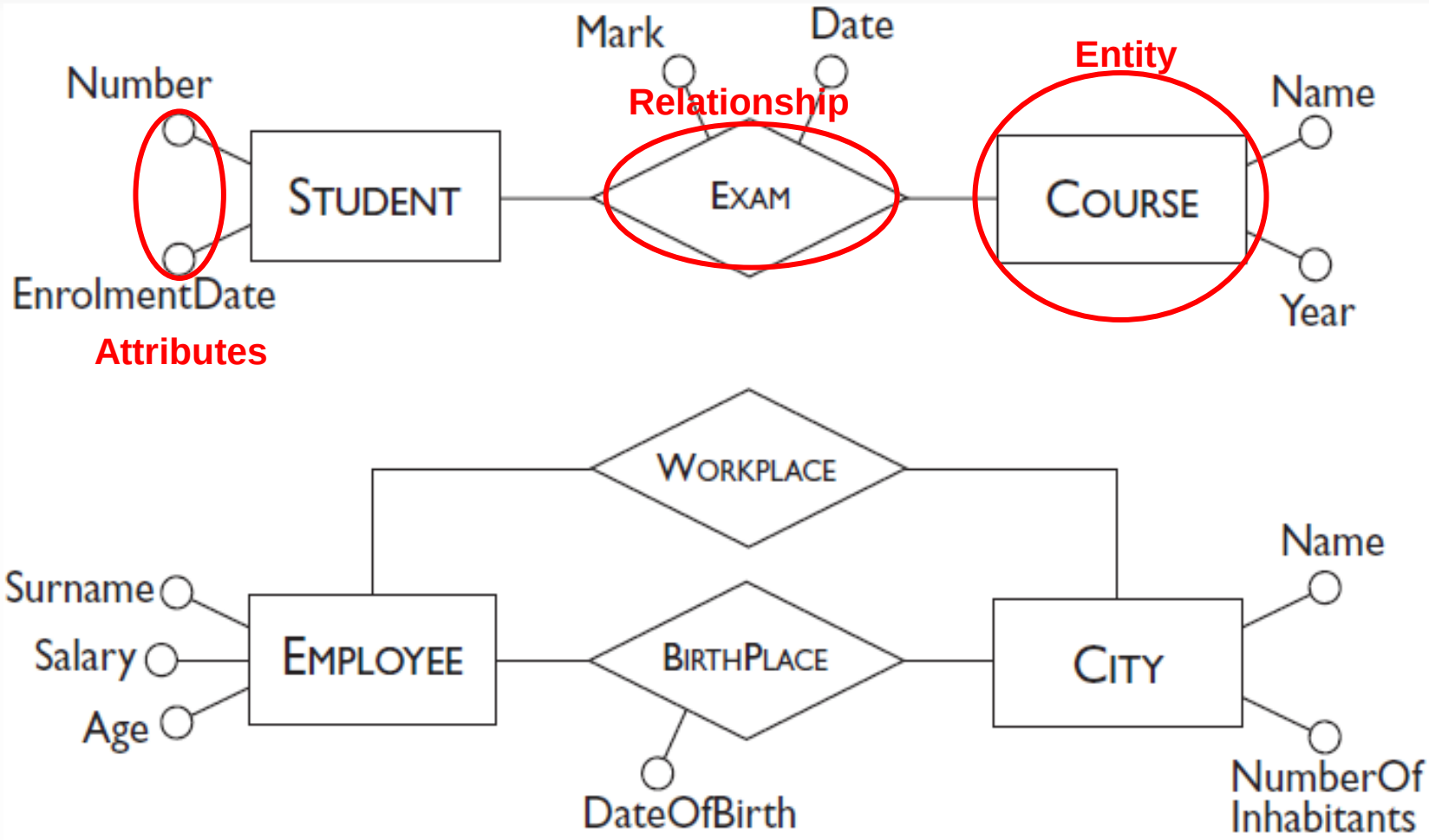
Example of occurrences of the SUPPLY relationship



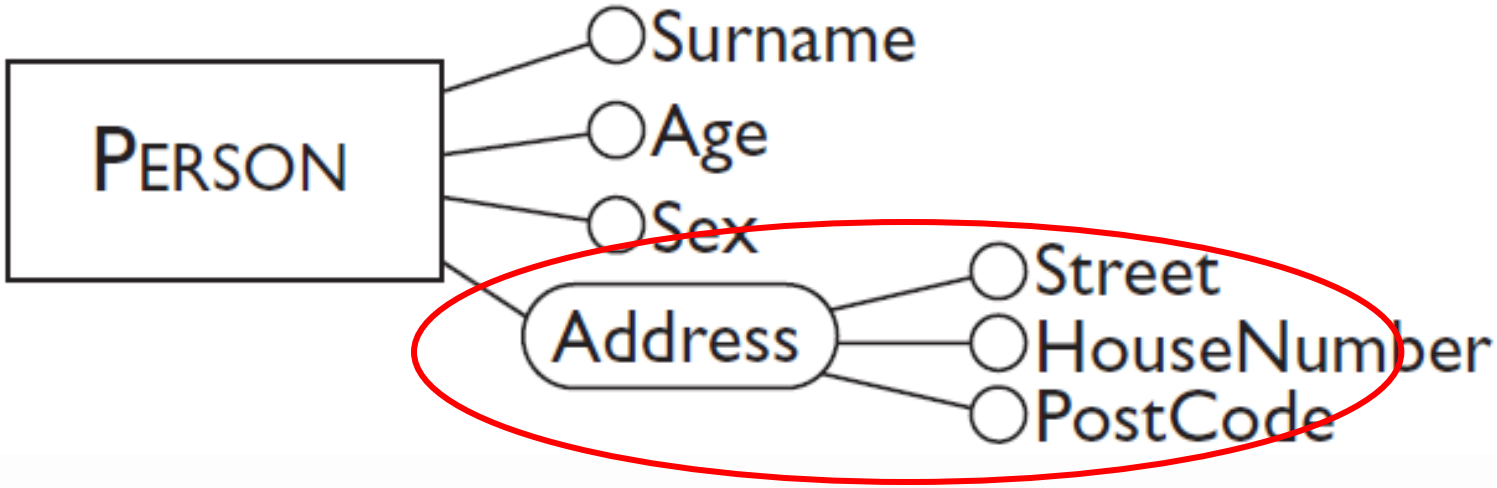
Attributes

- These describe the elementary properties of entities or relationships.
- **Surname**, **Salary** and **Age** are possible attributes of the **EMPLOYEE** entity, while **Date** and **Mark** are possible attributes for the relationship **EXAM** between **STUDENT** and **COURSE**.
- An attribute associates with each occurrence of an entity (or relationship) a value belonging to a set known as the domain of the attribute.
- The domain contains the allowable values for the attribute.

E-R schemas with relationships, entities and attributes

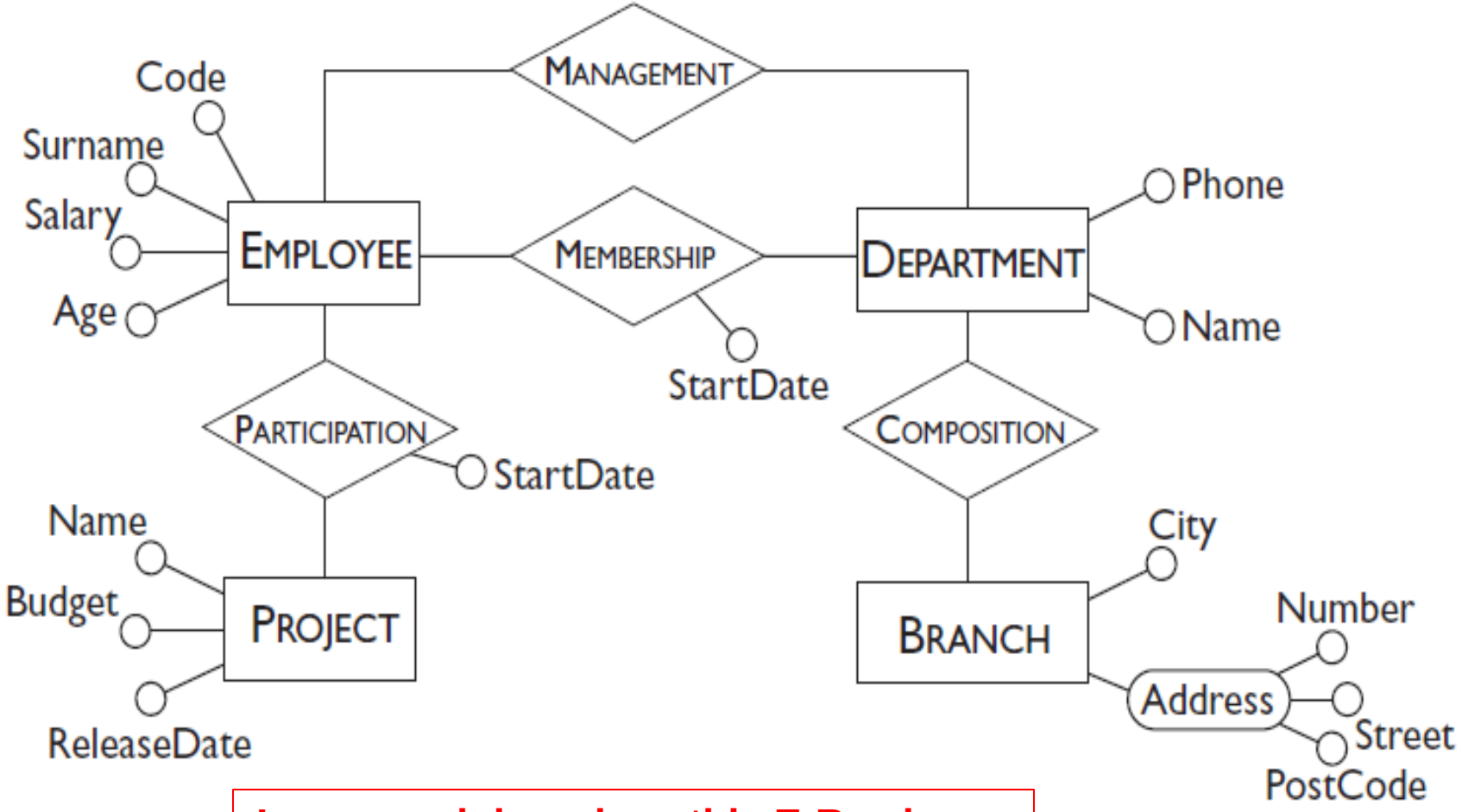


An example of an entity with **a composite attribute**



Composite attribute

An Entity-Relationship schema



In your opinion; does this E-R schema is complete?

Cardinalities

- They are specified for each entity participating in a relationship and describe the **maximum** and **minimum** number of relationship occurrences in which an entity occurrence can participate.

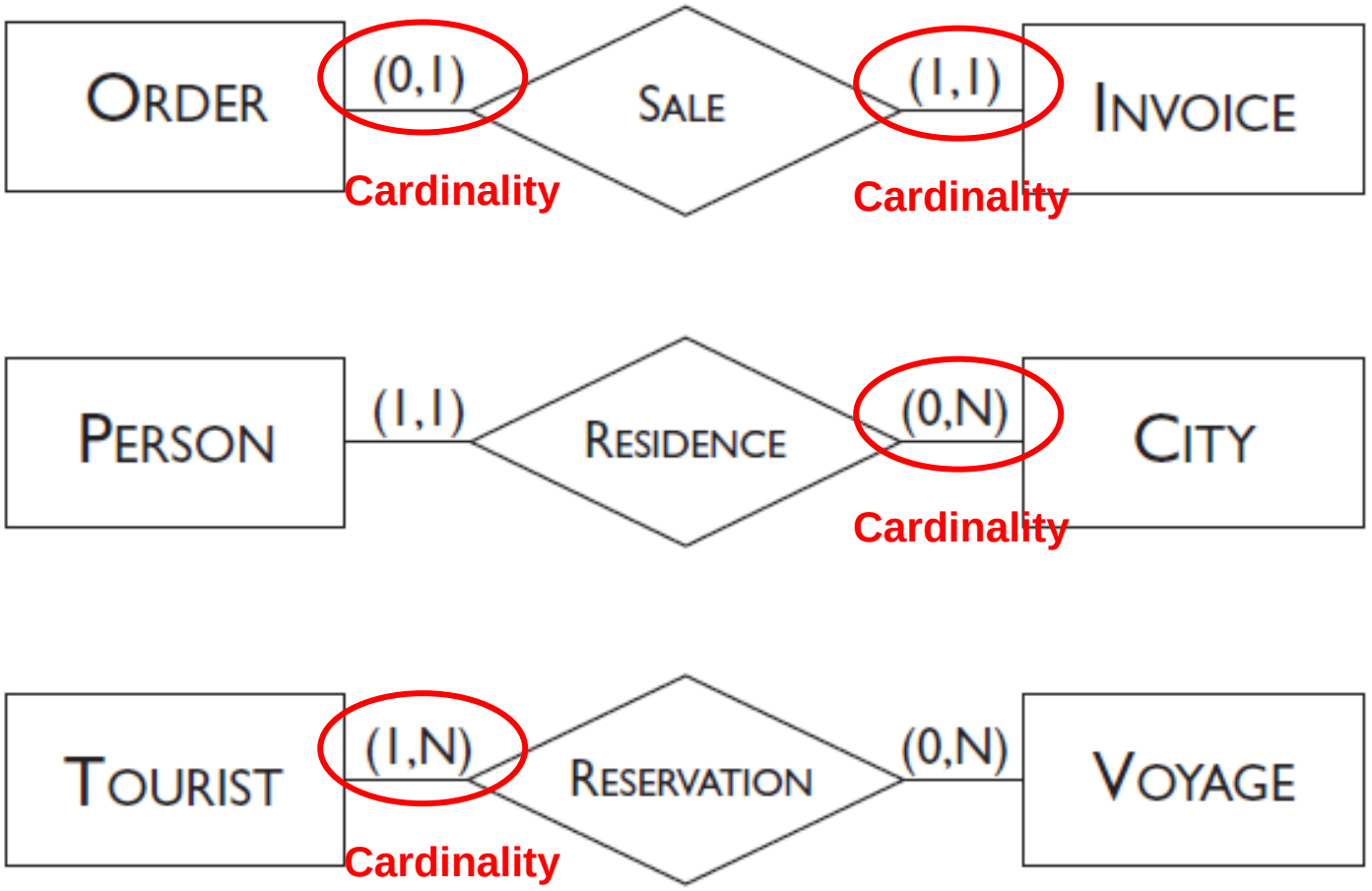
- They state therefore *how many times in a relationship between entities* an occurrence of one of these entities can be linked to occurrences of the other entities involved.

Cardinality of a relationship in the E-R model

Values for cardinalities

- In most cases, it is sufficient to use only three values for cardinalities: **zero**, **one** and the symbol **N**:
  - for the minimum cardinality, **zero** or **one**; in the first case we say that the participation in the relationship is ***optional***, in the second we say that the participation is ***mandatory***;
  - for the maximum cardinality, **one** or **many (N)**; in the first case each occurrence of the entity is associated at most with **a single occurrence** of the relationship, while in the second case each occurrence of the entity is associated with an arbitrary **number of occurrences** of the relationship.

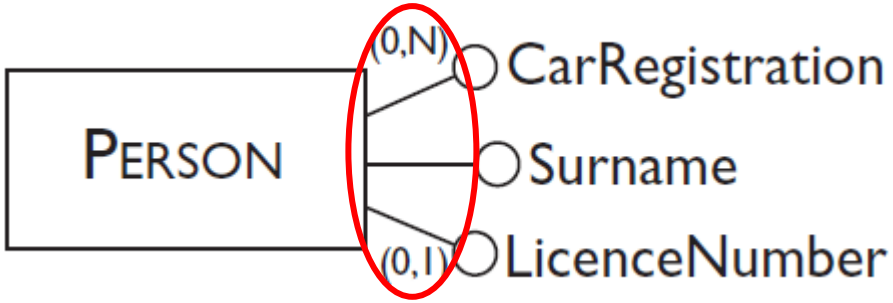
Examples of cardinality of a relationships



Cardinalities of attributes

- They can be specified for the attributes of entities (or relationships) and describe *the minimum and maximum number of values of the attribute associated with each occurrence of an entity or a relationship*.
- **In most cases,** the cardinality of an attribute is equal to (1,1) and is omitted.
- The value of a certain attribute can be null or there can exist various values of a certain attribute associated with an entity occurrence.

Examples of entity attributes with cardinality

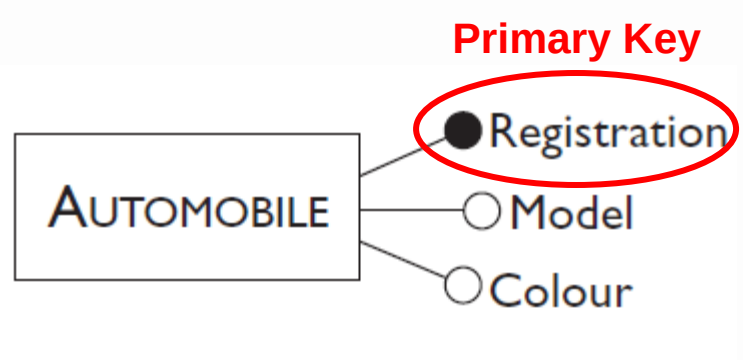


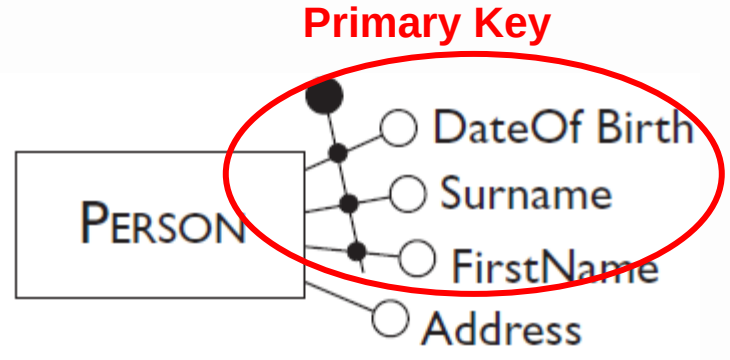
Cardinality on attributes

Identifiers

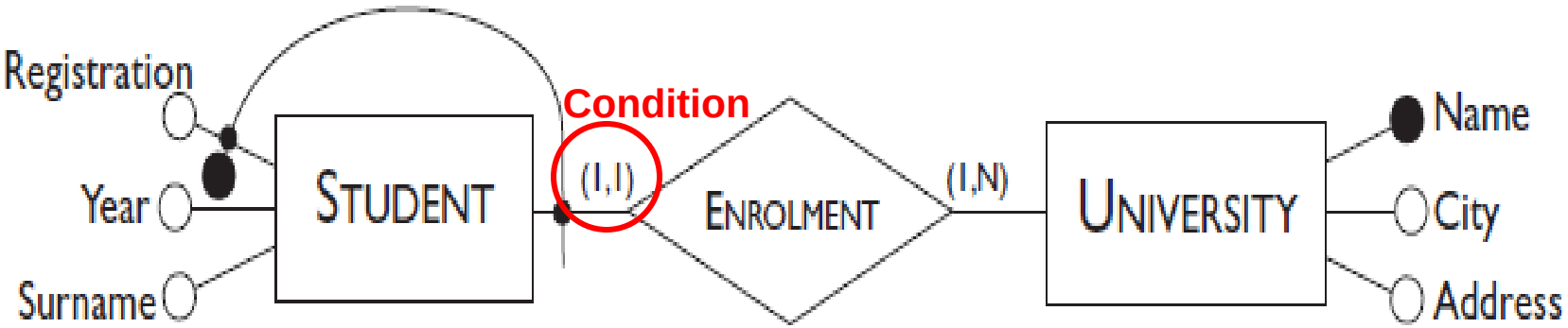
- They are specified for each entity of a schema and describe the concepts (attributes and/or entities) of the schema that allow the *unambiguous identification of the entity occurrences*.
- In many cases, an identifier is formed by **one** or **more** attributes of the entity itself: in this case we talk about an **internal identifier** (also known as a **key**).
- Sometimes, however, the attributes of an entity are **not** sufficient to identify its occurrences unambiguously and other entities need to be involved in the identification. This is called an **external identifier.**

Examples of internal identifiers





Examples of an external entity identifier



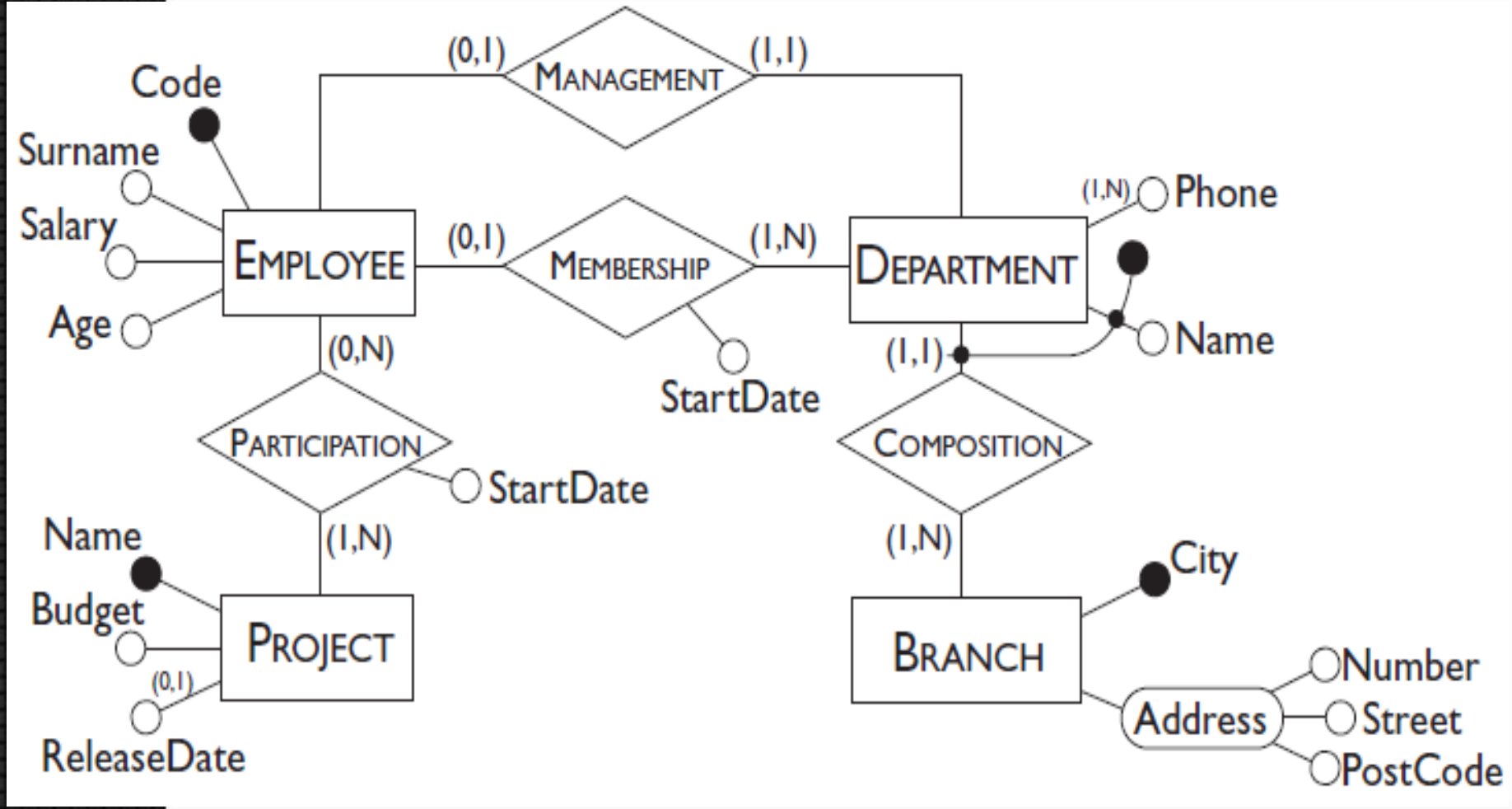
Name is the Primary Key for each university

Two students may have the same Registration in different universities

General observations on identifiers

- An identifier can involve one or more attributes, provided that each of them has **(1,1)** cardinality. **(Unique value)**
- An **external identifier** can involve one or more entities, provided that each of them is member of a relationship to which the entity to identify participates with **cardinality equal to (1,1).**
- Each entity must have at least one (internal or external) identifier.

A schema completed by identifiers and cardinality



Generalization

- These represent logical links between an entity E, known as parent entity, and one or more entities E1,...,En called child entities, of which E is more general, in the sense that it comprises them as a particular case.

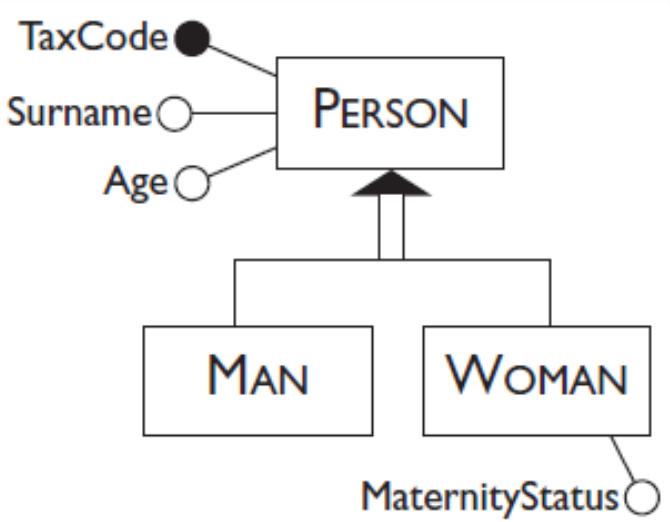
- In this situation we say that E is a generalization of E1,...,En and that the entities E1,...,En are specializations of the E entity.

Properties of Generalization

- Every *occurrence* of a child entity is also an occurrence of the parent entity.

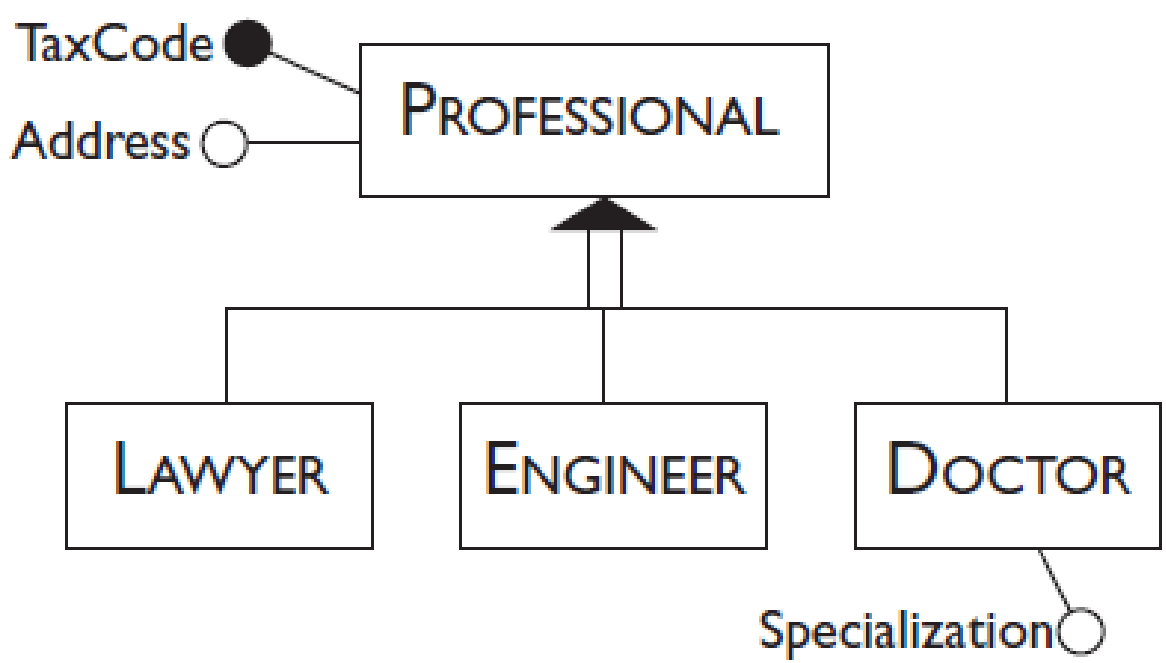
- Every *property* of the parent entity (attributes, identifiers, relationships and other generalizations) is also a property of a child entity. This property of generalizations is known as inheritance.

Examples of generalizations among entities



- PERSON is a generalization of MAN and WOMAN.
- MAN and WOMAN entities have the PERSON's attributes: TaxCode, Surname and Age.
- TaxCode is an identifier to MAN and WOMAN.

Examples of generalizations among entities

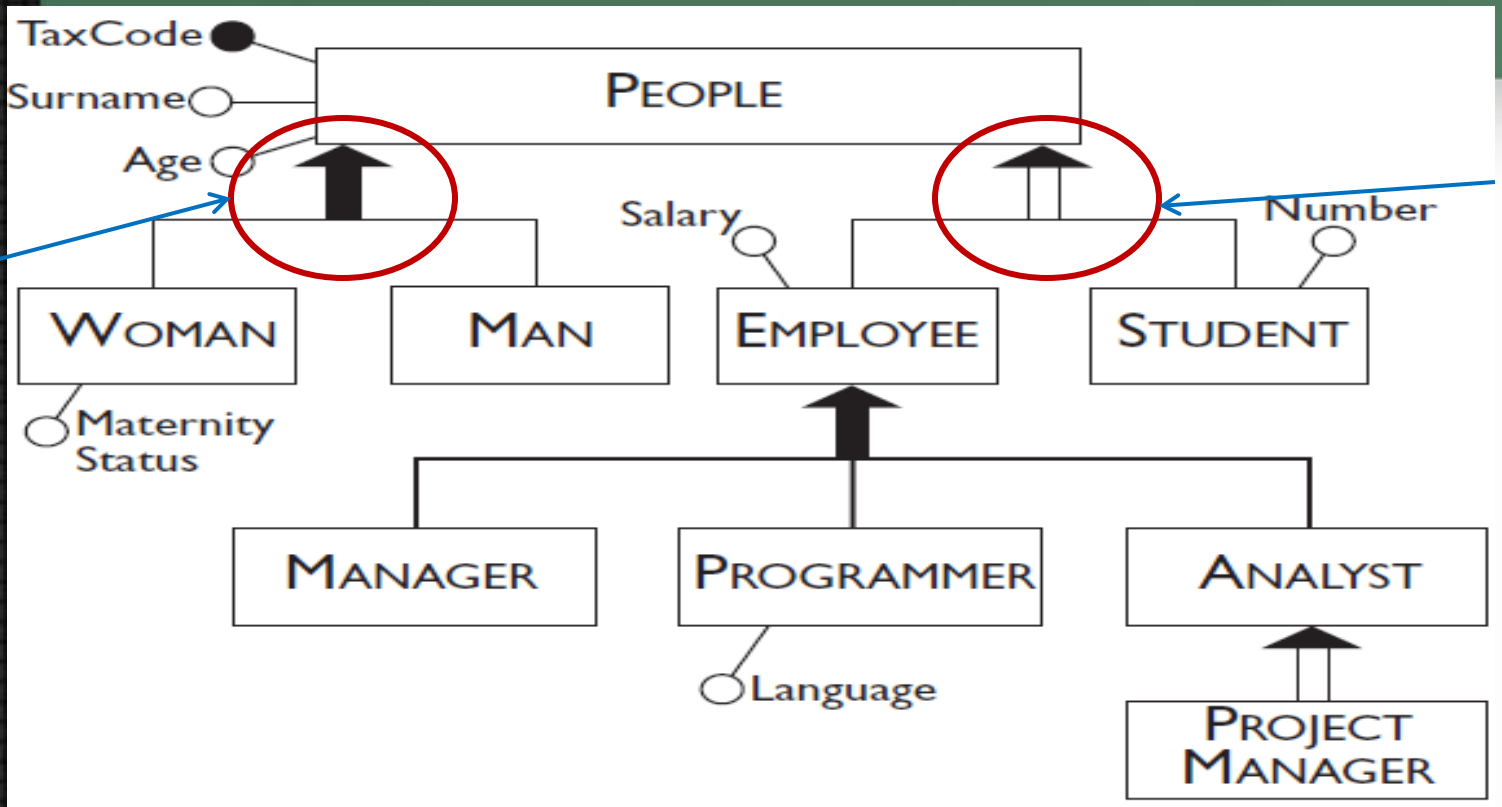


Classification of Generalization

- A generalization is ***total*** if every occurrence of the parent entity is also an occurrence of one of the child entities, otherwise it is ***partial***;

- A generalization is ***exclusive*** if every occurrence of the parent entity is at most an occurrence of one of the child entities, otherwise it is ***overlapping***.

Hierarchy of generalizations between entities



- **The generalization, PEOPLE, of MAN and WOMAN is:**
  - **Total (the men and women constitute 'all' the people)**
  - **Exclusive (a person is either a man or a woman)**
- **The generalization, PEOPLE, of EMPLOYEE and STUDENT is:**
  - **Partial and Overlapping, because there are students who are also employed.**

Report 1

Represent the following, using the constructs of the Entity-Relationship model:

1. In a zoological garden there are animals belonging to a species and having a certain age; each species is situated in a sector (having a name) of the zoo.
2. An automobile rental firm has a vehicle pool, in which each automobile has a registration number and a colour and belong to one category; for each category, there is a rental tariff.
3. A company produces CDs with a code and a title; each CD has been recorded by one or more singer, each of whom has a name and a address and some of whom have a stage name.

Report 2

Design an ER diagram.

Specify all key attributes and all structural constraints. A database is decided for the license issuing process of vehicles. The data requirements are as follows:

- The country is divided into departments (Cairo, Giza, Alex…etc). Each department is described by a code (unique), name (unique), and several service locations (e.g., for Cairo department, there are: Heliopolis, Nasr City, etc.).

Report 2….Continue

- Each vehicle is described by a number (unique for all vehicles in the same department), model, type(private, limousine, taxi,…etc), color, motor capacity, number of seats, manufacturing year, license issue date, license expiry date, owner, tax rate, and a set of fins. The owner, type, and tax rate information are mandatory for each vehicle. Each vehicle type is identified by a code (unique), name (unique), tax category has one or more vehicle types. Each tax category has a specific tax rate.

Report 2….Continue

- Each vehicle fin is described by a number (unique), type, date, and vehicle no. Each fin type is associated with a specific value.

- Each owner is described by a national number (unique), name, type (individual, organization, government, etc.), address, and set of phone numbers.

Thanks